On Random Betweenness Constraints

Andreas Goerdt

Technische Universität Chemnitz, Fakultät für Informatik
Straße der Nationen 62, 09107 Chemnitz, Germany
goerdt@informatik.tu-chemnitz.de
http://www.tu-chemnitz.de/informatik/TI/

Abstract. Ordering constraints are formally analogous to instances of the satisfiability problem in conjunctive normal form, but instead of a boolean assignment we consider a linear ordering of the variables in question. A clause becomes true given a linear ordering iff the relative ordering of its variables obeys the constraint considered.
The naturally arising satisfiability problems are **NP**-complete for many types of constraints. We look at random ordering constraints. Previous work of the author shows that there is a sharp unsatisfiability threshold for certain types of constraints. The value of the threshold however is essentially undetermined. We pursue the problem of approximating the precise value of the threshold. We show that random instances of the betweenness constraint (definition see Subsection 1.1) are satisfiable with high probability iff the number of randomly picked clauses is $\leq 0.92 \cdot n$, where n is the number of variables considered. This improves the previous bound which is $< 0.82 \cdot n$ random clauses. The proof is based on a binary relaxation of the betweenness constraint and involves some ideas not used before.

Keywords. Algorithms, logic, random structures, probabilistic analysis.

1 Introduction

1.1 Result

Let V always be a set of n variables. A 3-clause over V is an *ordered* 3-tuple (x, y, z) consisting of three different variables. Thus we have $n(n-1)(n-2) = (n)_3$ clauses altogether. A formula, also called ordering constraint is a set of clauses. Given a linear ordering of all n variables a clause evaluates to true if its variables satisfy a given constraint with respect to the ordering. A formula becomes true when all its clauses are true. This is the satisfiability problem the present paper deals with.

The clause (x, y, z) interpreted as a *betweenness* constraint is true iff y is between x and z, that is we have $x < y < z$ or $z <$

$y < x$ with respect to the ordering considered. The corresponding satisfiability problem is **NP**-complete [20].

We consider random ordering constraints interpreted as betweenness constraints. The random instance $F(V, m)$ or the corresponding probability space is obtained by picking a sequence (or set) of exactly m distinct clauses with uniform probability. Thus $F(V, m)$ is analogous to the well known random graph $G(n, m)$. More closely related to $F(V, m)$ are random 3-SAT formulas, see for example [3], [2] ,[9], [19]. The present paper is a successor to [13].

As common in the theory of random structures this paper deals with properties holding with high probability, that is $1 - o(1)$ when n becomes large and $m = m(n)$ is a given function. An additional piece of notation: A sequence of events E_n in some probability spaces holds with *uniformly positive probability* (abbreviated as *wupp*) if there is a constant $\varepsilon > 0$ such that $\text{Prob}[E_n] > \varepsilon$ for all sufficiently large n.

The probability space $F(V, p)$ is obtained by picking each clause independently with probability p. We call it the binomial space. For $p = r/n^2$ the expected number of clauses is rn. Moreover, the number of clauses is asymptotically equal to rn with high probability. Techniques as detailed on pages 34/35 of [4] show that the spaces $F(n, m)$ and $F(n, p)$ with $pn^3 = m = O(n)$ are for most questions of interest equivalent. This applies in particular to the satisfiability problems treated here as they are monotone problems. Following common usage we omit the technical details to show this each time.

The initial inspiration for the paper [13] came from some experiments (performed only for $n \leq 300$ for running time reasons.) These experiments show that the random betweenness constraint becomes unsatisfiable for rn random clauses when r is between 1.5 and 1.6. Results obtained in the cited paper collects

Fact 1. *For the random betweenness instance $F(V, m)$ with $m = rn$ the following events have high probability:*

(a) For $r \leq C$ the instance is satisfiable where $C < 0.82$.

(b) For $r > 4 \cdot \ln 2 \approx 2.77$ the instance is unsatisfiable.

(c) There exist numbers $C = C(n)$, $0.8 \leq C \leq 2.77$, such that for each constant $\varepsilon > 0$ we have unsatisfiability for $r \geq (C + \varepsilon)$ and satisfiability for $r \leq (C - \varepsilon)$.

Fact 1 (c) means that we have a sharp threshold for unsatisfiability, but we do not know the threshold value precisely. This is typical when the techniques from [11] are used. Given Fact 1 (a) it seems to be non-trivial to show that $F(V, m)$ is satisfiable with high probability for any m substantially larger than $0.82n$. We make some progress and prove

Theorem 2. *For $m = rn$ with constant $r \leq 0.921$ the random betweenness instance $F(V, m)$ is satisfiable wupp.*

Theorem 2 together with Fact 1 (c) implies a high probability result.

Corollary 3. *The random betweenness constraint with $m = rn$ and $r < 0.921$ is satisfiable with high probability.*

There are two different techniques to show that random structures are solvable (for example colourable in case of graphs or satisfiable in case of k-SAT instances:) On the one hand it has been successful to analyze heuristic algorithms and show that they find a solution to a random instance. On the other hand, and more recently non-constructive methods have been shown to be successful. In [1] it is shown that random k-SAT instances are satisfiable based on the second moment of the number of solutions and general probability estimates. Our proof consists of a first non-constructive part based on the second moment and a second constructive part.

1.2 More remarks the literature

Ordering constraints differ from traditional constraints like k-SAT or more general kinds of constraints in that the underlying assignment must be an *ordering* of all variables. This means on the one hand that each variable can receive one out of n values, its position in the ordering. On the other hand each of the n values can only be used once. Altogether we have $n! >> 2^n$ many assignments as opposed to only 2^n in the case of satisfiability.

Beyond random k-SAT there is a considerable body of work on random constraints with finite domain from which the values for each variable are taken. Only a small selection of the literature, in part due to Michael Molloy is [17], [18], [15]. The paper [17] points out that the investigation of thresholds is not only of structural interest, but has also algorithmic relevance: Random instances at thresholds often have some algorithmic hardness which makes them attractive as test cases for algorithms.

As far as we know systematic experimental studies of random ordering constraints have not been made. Our preliminary experiments indicate that instances closer to the threshold become harder. This shows that our study is relevant from the algorithmic point of view.

Ordering constraints tend to occur in knowledge representation formalisms. For example in [14] the cyclic ordering constraint occurs. In [7] a weighted version of an extended betweenness constraint is used to describe some biological situation. From the point of view of worst case complexity ordering constraints are investigated in [12] . Concerning the classification of the complexity of more general kinds of ordering constraints a recent breakthrough is [5] . We find [6] and [8] considering optimization versions of ordering constraints.

2 Outline of the proof of Theorem 2

We recall

Theorem 2. *For $m = rn$ with constant $r \leq 0.921$ the random betweenness instance $F(V, m)$ is satisfiable wupp.*

2-clause simply is a pair of distinct variables $x < y$, and we have $n(n-1) = (n)_2$ 2-clauses altogether. Given an ordering of the variables the 2-clause is satisfied iff x is smaller than y. A boolean assignment of the set of variables V is an assignment $a : V \to \{0, 1\}$ such that $n/2$ variables receive the value 1 and $n/2$ the value 0. Thus, in our case boolean assignments are *balanced.* A clause (x, y, z) is satisfied in the *boolean sense* by a iff it does *not* evaluate to $(0, 1, 0)$ or to $(1, 0, 1)$. Thus we have six out of 8 different possibilities to satisfy a clause in the boolean sense. A formula is satisfied in the boolean sense by a iff each clause is satisfied by a. A boolean assignment is equivalent to a partition of V into two sets V_0 and V_1 each with $n/2$ variables: V_0 the set of variables set to 0 and V_1 the set of variables set to 1. Let $A = y_1 < y_2 < y_3 < \ldots < y_n$. The ordering A induces the partition $V_l = \{y_1, \ldots, y_{n/2}\}$ and V_u is the upper half of the ordering. These notations directly imply

Proposition 4. *If the betweenness constraint F is satisfied by the ordering A then F is satisfied by the boolean assignment equivalent to the partition V_l and V_u.*

The following reduction allows to shrink a given betweenness constraint.

Definition 5. Let F be a betweenness constraint and let V_0, V_1 be a partition of V into two disjoint sets of $n/2$ variables each. If F is satisfied by the boolean assignment equivalent to V_0 and V_1 we say that the constraints F_0 over V_0 and F_1 over V_1 are defined.

Let (x, y, z) be a clause from F with at least two variables from V_0. It induces clauses as follows in F_0 :

- $(x, y, z) \in (V_0, V_0, V_0)(= V_0 \times V_0 \times V_0)$ implies $(x, y, z) \in F_0$
- $(x, y, z) \in (V_0, V_0, V_1)$ implies $x < y \in F_0$
- $(x, y, z) \in (V_1, V_0, V_0)$ implies $y > z \in F_0$.

Let (x, y, z) be a clause from F with at least two variables from V_1. It induces clauses as follows in F_1 :

- $(x, y, z) \in (V_1, V_1, V_1)$ implies $(x, y, z) \in F_1$,
- $(x, y, z) \in (V_1, V_1, V_0)$ implies $x > y \in F_1$,
- $(x, y, z) \in (V_0, V_1, V_1)$ implies $y < z \in F_1$.

F in the preceding definition has no clauses from (V_0, V_1, V_0) and (V_1, V_0, V_1) as it is satisfied by the boolean assignment associated to V_0 and V_1. The simple relationship between F and F_0 and F_1 is made clear by

Proposition 6. *Let F, V_0, and V_1 be such that F_0 and F_1 are defined. F is satisfied by a linear ordering with $V_0 < V_1$ iff F_0 and F_1 are both satisfiable.*

We consider the random instance $F = F(V, m)$. Given a boolean assignment a we define indicator random variables X_a and Y_a : $X_a(F) = 1$ if F is satisfied in the boolean sense by a. $X = \sum_a X_a$ is the number of satisfying boolean assignments. $Y_a(F) = 1$ iff F is satisfied by an ordering A which induces the *same partition* as a. (That is $V_l = V_0$ and $V_u = V_1$.) We let $Y = \sum_a Y_a$. The following remark follows from Proposition 4.

Remark 7. *(a)* $Y_a \leq X_a$.
(b) $Prob[F$ *is satisfiable as betweenness constraint* $] =$
$= Prob[Y \geq 1] \leq Prob[X \geq 1]$

We have $n!$ orderings as candidate solutions to a given betweenness constraint F. It is natural to consider the random variable which gives the number of satisfying orderings. However, in part due to the large number of $n! >> 2^n$ solution candidates this random variable seems not easy to deal with. The random variable Y is useful because it counts orderings associated to the same boolean assignment only once. It thus has to do only with 2^n candidates. The proof of the next proposition uses analytical techniques introduced in [1]. It is in Subsection 4.2

Proposition 8. *For the random instance* $F(V, m)$ *with* $m = rn, r \leq 1$ *we have:*
(a) $E[X] \geq (3/2)^{n(1-\varepsilon)}$ *for any constant* $\varepsilon > 0$.
(b) $E[X^2] \leq C \cdot (E[X])^2$ *for an appropriate constant* C.

As X is a random variable which is ≥ 0 and has finite variance we can use the Paley-Zygmund inequality: For any $0 \leq \Theta \leq 1$

$$\text{Prob}[X \geq \Theta E[X]] \geq (1-\Theta)^2 (E[X])^2 \;/\; E[X^2]. \tag{1}$$

With Proposition 8 as $E[X] \geq 1$ we directly get (but do not really need)

Corollary 9. *The event* $X \geq 1$ *holds wupp.*

Given a boolean assignment a, we consider the random instance $F_a(V, m)$ which is $F(V, m)$ conditioned on the event $X_a = 1$. Thus $F_a(V, m)$ consists of m clauses each satisfying the boolean assignment a. We have $b = (3/4)(n)_3 \cdot (1 + O(1/n))$ clauses satisfying a. The probability of a given instance of m such clauses is $(b)_m$ (in case of sequences of distinct clauses .) In the next section we prove the main

Lemma 10. *Let* a *be an arbitrary boolean assignment. We consider the random instance* $F_a(V, m)$ *with* $m = rn, r \leq 0.921$. *Then the event* $Y_a = 1$ *holds wupp.*

While Proposition 8 holds for $r > 0.921$, at present we cannot prove Lemma 10 for $r \geq 0.93$. Lemma 17 (b) gives the reason.

At this point the reader may wonder why we cannot derive Theorem 2 directly with Corollary 9 and the preceding Lemma. This however is not clear. The underlying probability spaces are not as closely related as it seems. In particular an instance from $F(V, m)$ with $X \geq 1$ may not be very random any more. It thus may not have much to do with a random instance $F_a(V, m)$ to which the Lemma refers. Instead we only use the second moment of X.

Proof of Theorem 2. For a suitable constant $\varepsilon > 0$ and any boolean assignment a we have with Lemma 10

$$\mathrm{E}[Y_a] = \mathrm{Prob}[X_a = 1] \cdot \mathrm{Prob}\ [Y_a = 1 | X_a = 1] \geq \ \mathrm{Prob}[X_a = 1] \cdot \varepsilon.$$

The second estimate above is Lemma 10. The first equation follows from the formula of total probability as Prob $[\, Y_a = 1 \,|\, X_a = 0\,] = 0$ (Remark 7 (a).) Then we get $\mathrm{E}Y \ \geq \varepsilon {\cdot} \mathrm{E}X \to \infty$ (with Prop. 8 (a).)

Furthermore we have

$$\begin{aligned} \mathrm{E}[Y^2] \ &\leq \sum_{(a,b)} \mathrm{Prob}[X_a = 1 \text{ and } X_b = 1] \\ &= \mathrm{E}[X^2] \leq C(\mathrm{E}X)^2 \ \leq (C/\varepsilon^2)(\mathrm{E}Y)^2 \end{aligned}$$

using Remark 7 (a) for the first estimate and Proposition 8 (b) to bound $\mathrm{E}[X^2]$. Now, Theorem 2 follows with Equation (1). □

3 Proof of Lemma 10

We switch to the binomial space because the subsequent probability calculations appear slightly easier. The random instance $F(n, p, q)$ is obtained by throwing each 3-clause randomly with p and each 2-clause with q. We let

$$c = (1/3) \cdot r \text{ and } d = (2/3) \cdot r, \ \text{ or } r = 3c = (3/2)d$$

for the rest of this section. The main work is to prove

Lemma 11. *For $r = 0.921$ $F(n, c/n^2, d/n)$ is satisfiable wupp.*

Proof of Lemma 10 from Lemma 11 . For $F = F_a(V, m)$, we have $Y_a(F) = 1$ iff both formulas F_0 and F_1 as in Definition 5 are satisfiable (Proposition 6, $V_0 = a^{-1}(0)$ and $V_1 = a^{-1}(1)$).

In F the number of clauses from each of the 6 admissable possibilities among (V_i, V_k, V_j) with $i, j, k = 0, 1$ is concentrated at its expectation, that is asymptotically $(1/6)m$ with high probability. With high probability 2 clauses which overlap in 2 variables do not occur as m is linear in n. Therefore F_0 and F_1 have $(1/6)m$ many 3-clauses and $(2/6)m$ many 2-clauses each over $n/2$ variables. Moreover, F_0 and F_1 are stochastically independent (given their respective number of clauses which is concentrated). For $m = rn$ F_0 and F_1 are two independent random formulas with asymptotically cn 3-clauses and dn 2-clauses over n variables (scaling to n variables instead of $n/2$.)

Following the remark in the Introduction concerning the binomial space, $F(n, c/n^2, d/n)$ is satisfiable wupp implies that F_0 and F_1 are both satisfiable wupp, by independence of F_0 and F_1. Lemma 10 follows. $\square$

Definition 12. The directed (multi-)graph of F has as vertices the variables of F. Its edges are given by: The clause $C = (x, y, z) \in F$ induces the edges $(x, y), (y, z)$ and (x, z) each marked with C. The clause $x < y \in F$ induces the edge (x, y).

Clearly, if the graph of F is cycle free then F is satisfiable (by any topological ordering of the graph.) To reduce a formula F we apply

Algorithm 13. Input: A formula F.
V_1 := the set of those variables which occur exactly once in a 3-clause of F and nowhere else.
V_2 := those variables x which occur *only* at the position $x < -$. This means that all 2-clauses with x are of the form $x < y$ and we have no 3-clauses with x. Here the case that x does not occur at all is included.
V_3 := the variables x which occur *only* and at least once as $- < x$.

The result H of the algorithm is obtained by deleting all variables from $V_1 \cup V_2 \cup V_3$ and clauses containing them from F.

The algorithm is correct in the sense of

Lemma 14. *If H is satisfiable then F is satisfiable.*

We iterate Algorithm 13 and therefore need

Definition 15. The $w_{2,k}$ and $w_{3,k}$ for $k \geq 0$ are defined inductively by

$$\begin{aligned} w_{2,0} &= w_{3,0} = 0, \\ w_{2,k+1} &= \exp\left(-d(1-w_{2,k})\right) \cdot \exp\left(-3c(1-w_{3,k})^2\right), \\ w_{3,k+1} &= \exp\left(-d(1-w_{2,k})\right) \cdot w_{2,k+1}. \end{aligned}$$

Following [19] we consider the following type of random Poisson hypertree (which is approximately the random neighbourhood of a given variable in $F(n, c/n^2, d/n)$.) The vertices of the tree are variables and edges are clauses. For each variable in the tree we have $<$-clauses resp. $>$-clauses, and 3-clauses as candidate child clauses. The number of child $<$- resp. $>$-clauses is distributed according to $\mathrm{Po}(d)$, the Poisson distribution with parameter d. The number of child 3-clauses follows $\mathrm{Po}(3c)$. All distributions are independent. The random hypertree of depth l is obtained by generating l generations of children starting from a given root variable.

For the sake of analyzability we apply Algorithm 13 to the random hypertree of depth l in the following way: In the first iteration we apply the algorithm to the variables in depth $l-1$, in the second iteration to the variables in depth $l-2$, and so on.

Lemma 16. *Let $1 \leq k \leq l-1$. Conditional on the event that x is a variable in depth $l-k$ of the random hypertree of depth l the probability that x gets deleted in the k'th iteration is*

(a) $w_{2,k}$ if x is connected to its father by a 2-clause,
(b) $w_{3,k}$ if x is connected to its father by a 3-clause.

Proof. By induction on k. For the induction step we proceed as follows. Condition on the event that x is a variable in depth $l-k$. After the $k-1$'st iteration of the algorithm we have: The probability that

x has no child $<$-clauses is

$$\sum_{m\geq 0} \text{Prob}[\text{Po}(d)=m]\cdot w_{2,k-1}^m = \sum_{m\geq 0} \frac{(dw_{2,k-1})^m}{m!}\cdot \exp(-d)$$
$$= \exp(-d(1-w_{2,k-1})).$$

A similar calculation shows that the probability that x has no child 3-clauses is $\exp\left(-3c(1-w_{3,k-1})^2\right)$, multiplying we get $w_{2,k}$. We proceed for $w_{3,k}$ in the same way. □

Lemma 17. *(a) For* $r = 0.921$ *and* $k \to \infty$ *we have* $w_{2,k}, w_{3,k} \to 1$.
(b) For $r = 0.93$ *we have* $w_{2,k} < 0.605$ *and* $w_{3,k} < 0.475$ *for all* k.

We postpone the proof of Lemma 17 to Subsection 4.1.

Lemma 18. *Let* $r = 0.921$ *and let* k *be such that* $w_{3,k} > 0.95$. *We denote* $S := 3 \cdot \ln n$. *Let* H *be the formula obtained after iterating Algorithm 13* k*-times starting with* $F = F(n, c/n^2, d/n)$.

(a) The expected number of cycles of length $\geq S+1$ *in* H *is* $o(1)$.
(b) The expected number of cycles of length $2 \leq s \leq S$ *in* H *is*

$$\frac{(3c(1-w_{3,k})+d)^s}{s} + O(1/n^{0.7}).$$

With Lemma 18 we can finish the main argument:

Proof of Lemma 11 . Let $F = F(n, c/n^2, d/n)$ and let H be the formula obtained after k iterations of Algorithm 13 from F. We use the k from Lemma 18. The probability to have a cycle of length $\geq S+1$ in H is $o(1)$ by (a) of Lemma 18.

The probability to have a cycle of any length between 2 and S is bounded above the expectation. By (b) of Lemma 18 we have a bound of

$$\sum_{s=2}^{S} \frac{(3c(1-w_{3,k})+d)^s}{s} + S\cdot O(1/n^{0.7})$$
$$< (0.7)^2/2 \cdot \sum_{s\geq 0} (0.7)^s + o(1)$$
$$= (0.49\cdot 10)/(2\cdot 3) + o(1) < 1,$$

where we use that $3c(1 - w_{3,k}) + d < 0.7$ and the geometric series.

H has no cycle wupp. With the remark after Definition 12 H is satisfiable. Using Lemma 14 inductively we get that F itself is satisfiable wupp. $\square$

Proof of Lemma 18. First the values:

$$3c = 0.921, \quad d = 0.614, \quad 3c + d = 1.535 \quad \text{and} \quad 3c/d = 1.5.$$

Occasionally we assume that constants like c and d are slightly larger than the original ones, this allows us to treat expressions like $(c + o(1))^n$ simply as being bounded above by c^n.

We consider only cycles $x_0 \to x_1 \to x_2 \cdots \to x_0$ such that the x_i are distinct (simple cycles). Moreover, we assume wlog. that different edges $x_i \to x_{i+1}$ are induced by different clauses. Otherwise we have a 3-clause (x, y, z) and the piece $\cdots x \to y \to z \cdots$ on the cycle. This piece is replaced with $x \to z$.

A given edge is induced by one 2-clause and $3(n-2)$ 3-clauses. Therefore the expected number of ways in which an edge occurs in the graph of F is $3(n-2)c/n^2 + d/n \;=\; 1.535/n$. The probability that a given edge is present is
$1-(1-c/n^2)^{3(n-2)} \cdot (1-d/n) = (3c+d)/n + O(1/n^2)$. Observe that $(3c + d) = 1.535/n$. Disregarding dependencies between edges F induces a directed random graph well above the strongly connected component threshold which occurs at edge probability $1/n$, see [21], [16]. As such F itself should contain cycles.

Proof of Lemma 18 (a). Each cycle of length $\geq S+1$ contains a simple path of length S. We show that the expected number of such paths in H is $o(1)$. Let $x_0 \to x_1 \to x_2 \cdots \to x_S$ be a candidate path with t edges in t fixed slots induced by 3-clauses. The additional variables to fill the 3-clauses are denoted by $(y_1, \ldots, y_t)$. We assume that y_i includes the information about its slot (one out of 3) in its 3-clause. Altogether we have $< n^{S+1} \cdot \binom{S}{t} \cdot (3n)^t$ such candidate paths. The probability that a candidate path occurs in F is $(c/n^2)^t \cdot (d/n)^{S-t}$.

First case: Let C be a constant. In F the expected number of paths with $t \leq C$ is bounded above as

$$n^{S+1} \cdot \sum_{t \leq C} S^t \cdot (3n)^t \cdot (c/n^2)^t \cdot (d/n)^{S-t}$$
$$< O(\text{ poly } (\log n)) \cdot n \cdot (3c/d)^C \cdot d^S \; = \; o(1)$$

note that $3c/d = 1.5 (> 1)$, $3 \ln d \approx -1.429$, and $S = 3 \ln n$.

Second case: We consider paths with $t > C$ such that the number of *new* variables among $(y_1, \ldots, y_t)$ is $\leq t - 3$. A y_i is not new if it occurs among the x_j or there is a $j < i$ with $y_j = y_i$. For the expected number of paths of this type in F we get an upper bound of

$$n^{S+1} \cdot \sum_{t > C} \binom{S}{t} S^3 \cdot (3n)^{t-3} \cdot (3 \cdot \text{ poly } (\log n))^3 \cdot (c/n^2)^t \cdot (d/n)^{S-t}$$
$$< n \cdot \text{ poly } (\log n) \cdot (3c + d)^S \cdot (1/n)^3 \; = \; o(1)$$

as $3 \ln(3c + d) \approx 1.28$ and $S = 3 \ln n$.

For candidate paths to which the preceding two cases do not apply we consider the neighbourhood of those y_i which occur only once in the candidate. There are at least $t - 4$ of them as there are at least $t - 2$ new variables among $(y_1, \ldots, y_t)$.

The following principle is our guide: Let $y = y_j$ be a variable occurring only once and let $x_i \to y \to x_{i+1}$ be the 3-clause with y. If k iterations of Algorithm 13 have the effect that y is deleted the edge $x_i \to x_{i+1}$ disappears, too. A lower bound on the probability of this event can be derived from that part of the k-neighbourhood of y which is still random. This means the standard k-neighbourhood of y is modified in that the clause $x_i \to y \to x_{i+1}$ is not any more included. (The 1-neighbourhood of y in our sense are the clauses containing y, except of $x_i \to y \to x_{i+1}$.)

Let P be a given candidate path, let $y_1, \ldots, y_{t-4}$ be variables among the y_i which occur only once. The k-neighbourhood of $y_1, \ldots, y_{t-4}$ altogether is the union of the k-neighbourhoods of $y_1, \ldots, y_{t-4}$. We call the k-neighbourhood of $y_1, \ldots, y_{t-4}$ *exceptional* if the number of y_i whose k-neighbourhoods are hypertrees disjoint from the rest of the k-neighbourhood is $\leq t - 4 - 6 = t - 10$. Let P_{EX} be the event that the path P occurs in F and that the k-neighbourhood of $y_1 \dots y_{t-4}$ is exceptional. Further below we prove the following

Claim. The probability of P_{EX} is bounded above as

$$(c/n^2)^t \cdot (n/d)^{S-t} \cdot O(\text{ poly } (\log n)/n)^3.$$

Third case: The expectation of the number of paths P in F for which the event P_{EX} holds is bounded above as

$$\begin{aligned} n^{S+1} \cdot \sum_{t>C} \binom{S}{t} \cdot (3n)^t \cdot (c/n^2)^t \cdot (d/n)^{S-t} \cdot O(\text{ poly } (\log n))^3 \\ < n \cdot (3c+d)^S \cdot O(\text{ poly } (\log n)/n)^3 = o(1) \end{aligned}$$

as in the second case.

Fourth case: Let P be a candidate with t edges induced by 3-clauses and at least $t-4$ new variables. The probability that P is present in H and $P_{\neg EX}$ in F is

$$\begin{aligned} &\text{Prob } [P \text{ in } H \wedge P_{\neg EX}] \\ &= \quad \text{Prob } [P \text{ in } H \mid P_{\neg EX}] \cdot \text{ Prob } [P_{\neg EX}] \\ &< \quad w^{t-9} \cdot (c/n^2)^t \cdot (n/d)^{S-t}. \end{aligned} \tag{2}$$

Estimate (2) holds because of the following observations.

Prob $[P_{\neg EX}] < (c/n^2)^t \cdot (n/d)^{S-t}$ by definition of $P_{\neg EX}$. Conditional on $P_{\neg EX}$ we can assume wlog. that the k-neighbourhood of $y_1, \dots, y_{t-9}$ consists of hypertrees disjoint from the rest of the k-neighbourhood. This part of the neighbourhood can be viewed as $t-9$ independent Poisson hypertrees as considered in Lemma 16. (We skip the detailed argument on this point and refer to [19].) By Lemma 16 (b) the probability that y_i survives in H is $< w$, independently

For the expected number of paths P with $P_{\neg EX}$ in F we therefore get a bound of

$$\begin{aligned}
&n^{S+1} \cdot \sum_{t>C} \binom{S}{t} \cdot (3n)^t \cdot w^{t-9} \cdot (c/n^2)^t \cdot (d/n)^{S-t} \\
&< n \cdot \sum_{t>C} \binom{S}{t-9} \cdot (3cw)^{t-9} \cdot d^{S-(t-9)} \cdot (S \cdot 3c/d)^9 \\
&\qquad < n \cdot \text{poly } (\log n) \cdot (3cw + d)^S = o(1)
\end{aligned}$$

as $3\ln(3cw+d) \approx -1.234$ and $S = 3\ln n$.

Summing the preceding four $o(1)$-terms we get that the expected number of paths of length S in H is $o(1)$.

Proof of the claim. The claim is easily proved by generating the k-neighbourhood step by step. Conditionings on appropriate events (which we skip) would lead to a complete formalization. Recall that $y_1, \ldots, y_{t-4}$ are the additional variables of the 3-clauses of P which occur only once. We first generate the neighbours of $y = y_1$. The 3-clauses with y follow the binomial distribution $\text{Bin}(3(n-1)(n-2)-1, c/n^2)$, for the 2-clauses we get two $\text{Bin}(n-1, d/n)$ distributions. These distribution are independent as they refer to disjoint sets of clauses. The probability that we get a clause which produces an overlap (that is it contains another variable of the path P or collides with another neighbor of y is $O(\text{ poly}(\log n)/n)$. Observe that the probability to get at least $a \log n$ new 3-clauses with y (a a small constant) is very small:

$$\sum_{h \geq a \log n} \binom{3n^2}{h} \left(\frac{c}{n^2}\right)^h \leq \frac{(3c)^{a \log n}}{(a \log n)!} \cdot \sum_{h \geq 0} \frac{(3c)^h}{h!} \;<\; 1/n^{\Omega(\log \log n)}.$$

The same argument applies when we consider the 2-clause distribution. When we have $> a \log n$ new clauses we stop the process. We condition the following consideration on the event that this has not happened.

When we have an overlap by now we proceed with the next new y_i which still occurs only once (there are at least $t-4-2$ of them Otherwise we generate the next generation of neighbours of y. Due to our conditioning the distributions stay essentially unchanged and we can continue analogously. When we have generated k generations of neighbours of y without overlap we go to y_2. We proceed in this way and stop only when we have 3 overlaps or a variable with degree $> a \log n$.

The probability to hit upon 3 overlaps is $O(\text{ poly } (\log n)/n^3)$ as long as the number of neighbours of each variable is $< a \log n$. This is ensured by our conditioning. Moreover, the probability of $> a \log n$ neighbours at some point is much smaller than $O(\text{ poly } (\log n)/n^3)$ and thus can be added without weakening the upper bound. This proves the claim because when the number of independent hypertrees rooted at the y_i is reduced from $t-4$ to $t-10$ we must have at least 3 overlaps in the neighbourhood.

Proof of Lemma 18 (b). We refer to (a) for unexplained notions and employ the same principle. Let $x_0 \to x_1 \to \ldots \to x_{s-1} \to x_0$ be a candidate cycle of length s with $s \geq t \geq 0$ edges induced by 3-clauses. Let $(y_1, \ldots, y_t)$ be the additional variables of the 3-clauses (each variable comes with its slot in the 3-clause, 1 out of 3.) The number of such candidates is bounded above as $(n^s/s) \cdot \binom{s}{t} \cdot (3n)^t$. (The $1/s$ because of cyclic permutations.) The probability that a candidate occurs in F is $(c/n^2)^t \cdot (d/n)^{S-t}$. The expectation of the number of cycles of length s in F is bounded above by $(3c+d)^s/s$. Recall that $3c+d = 1.535$. We bound the expected number of cylces in H depending on the overlap structure among the y_i and the k-neighbourhood in F.

First case: We consider cycles of length s such that all y_i are new.

Conditioning on the event that the k-neighbourhood in F consists of disjoint hypertrees each y_i surives with probability bounded above by w independently. For the expected number of cycles whose the k-neighbourhood in F consists of disjoint hyperterees and which survive in H we get a bound of $(3cw+d)^s/s$.

The event that we have exactly one overlap in the k-neighbourhood has probability $O(\log n/n)$. Conditioning in this event, at least $t-2$ of the y_i survive with probability bounded above by w independently. For the expected number of cycles in H with such a k-neighbourhood in F we get a bound of

$$n^s \cdot \sum_{t \geq 0} \binom{s}{t} \cdot (3n)^t \cdot (c/n^2)^t \cdot (d/n)^{s-t} \cdot w^{t-2} \cdot O(\text{ poly } (\log n)/n)$$
$$< (3cw + d)^s/s \cdot O(\text{ poly } (\log n)/n) < O(\text{ poly } (\log n)/n).$$

For two or more overlaps in the k-neighbourhood we need only F and bound the expectation as

$$(3c + d)^s/s \cdot O(\text{ poly } (\log n)/n)^2 = O(1/n^0.7)$$

as $s \leq S = 3 \ln n$ and $3 \ln(3c + d) \approx 1.28$.

Second case: If one y_i is not the number of candidates is $n^s \cdot (3n)^{t-1} \cdot O(\text{ poly } (\log n))$. If we have no overlap in the k-neighbourhood we have at least $t - 2$ independent hypertrees rooted at the y_i. We get the same result as in the first case with one overlap.

If we have in addition one overlap in the k-neighbourhood we have another factor of $O(\text{ poly } (\log n)/n)$. We get the same result as in the first case with at least 2 overlaps.

Third case: If at least two y_i are not new we have $n^s \cdot (3n)^{t-2} \cdot O(\text{ poly } (\log n))^2$ candidates and get the same result as the first case with at least two overlaps. □

4 The remaining proofs

4.1 Proof of Lemma 17

Recall

Lemma 17. *(a) For* $r = 0.921$ *and* $k \to \infty$ *we have* $w_{2,k}, w_{3,k} \to 1$. *(b) For* $r = 0.93$ *we have* $w_{2,k} < 0.605$ *and* $w_{3,k} < 0.475$ *for all* k.

Proof. We define

$$w_0 = 0, \quad w_1 = \exp(-d) \cdot \exp(-3c) \quad \text{and for } k \geq 1$$
$$w_{k+1} = \exp\Big(-d(1-w_k)\Big) \cdot \exp\left(-3c \cdot \Big(1 - \underbrace{\exp\Big(-d(1-w_{k-1})\Big) \cdot w_k}_{\text{Compare to } w_{3,k}}\Big)^2\right)$$

By induction $w_k = w_{2,k}$. Moreover, $0 \leq w_k \leq 1$, $w_{k+1}/w_k > 1$ and the w_k are strictly monotonously increasing. Therefore they have a limit $W := \lim w_k$. We have $0 < W \leq 1$.

The recursive definition of w_{k+1} induces the function

$$F(w) := \exp\Big(-d(1-w)\Big) \cdot \exp\left(-3c \cdot \left(1 - \overbrace{\underbrace{\exp\Big(-d(1-w)\Big)}_{G(w):=} \cdot w}^{K(w) := G(w)w =}\right)^2\right).$$

By continuity $F(W) = W$, that is W is a fixpoint of $F(w)$. We recall that $r = 3c = (3/2)d$ and rewrite

$$F(w) = \exp\left(\underbrace{d \cdot \Big(-(5/2) + w + 3K(w) - (3/2) \cdot (K(w))^2\Big)}_{E(w):=}\right).$$

Some derivatives:

$$\begin{aligned}
G'(w) &= d \cdot G(w) \ \text{ and } \ G''(w) = d^2 \cdot G(w) \\
K'(w) &= d \cdot K(w) + G(w) = (1 + dw)\, G(w) \\
K''(w) &= d \cdot G(w) + d \cdot (1 + dw) \cdot G(w) = d^2 \cdot K(w) + 2d \cdot G(w) \\
E'(w) &= d \cdot \Big(1 + 3K'(w) - \frac{3}{2} \cdot 2 \cdot K(w) \cdot K'(w)\Big) \\
&= d \cdot \Big(1 + 3K'(w) \cdot (1 - K(w))\Big) \\
&= d \cdot \Big(1 + 3 \cdot G(w) \cdot (1 + dw) \cdot (1 - K(w))\Big)
\end{aligned}$$

Some facts for $0 \leq w \leq 1$:

- $G(w) > 0, G'(w) > 0, G''(w) > 0$.
- $G(w)$ strictly monotonously increasing from $G(0) = \exp(-d)$ to $G(1) = 1$. $G(w)$ is convex.

- $K(w) \geq 0\,, K'(w) > 0,\; K''(w) > 0\,.$
- $K(w)$ is strictly monotonously increasing from $K(0) = 0$ to $K(1) = 1$. $K(w) \leq G(w)$ and $K(w)$ is convex.
- $E(w)$ is strictly monotonously increasing from $E(0) = -(5/2)d$ to $E(1) = 0$.
- By the preceding item $F(w)$ is strictly monotonously increasing from $F(0) = \exp(-5d/2)$ to $F(1) = 1$.
- W is the *smallest* positive fixpoint of $F(w)$ because of monotonicity.

  For $w > 0$ we have

$$F(w) = w \Leftrightarrow \underbrace{F(w)/w}_{H(w) :=} = 1.$$

As $H(w) \to +\infty$ for $w \to 0$ and $H(1) = 1$, there is positive fixpoint $w < 1$ of $F(w)$ iff $H(w)$ has a local minimum ≤ 1 in $0 < w < 1$.

$$\begin{aligned} H'(w) &= \frac{F(w) \cdot (E'(w) \cdot w - 1)}{w^2} \\ E'(w) \cdot w &= d \cdot \Big(w + 3 \cdot K(w) \cdot (1 + dw) \cdot (1 - K(w))\Big) \qquad (3) \end{aligned}$$

using $wG(w) = K(w)$ to get Equation (3).

$$\begin{aligned} H'(w) &\to -\infty \text{ for } w \to 0 \\ H'(w) < 0 \Leftrightarrow 1 > E'(w) \cdot w &\Leftrightarrow \frac{1 - dw}{1 + dw} > 3d \cdot K(w) \cdot (1 - K(w)) \\ H'(w) > 0 \Leftrightarrow 1 < E'(w) \cdot w &\Leftrightarrow \frac{1 - dw}{1 + dw} < 3d \cdot K(w) \cdot (1 - K(w)) \qquad (4) \\ H'(w) = 0 \Leftrightarrow 1 = E'(w) \cdot w &\Leftrightarrow \underbrace{\frac{1 - dw}{1 + dw}}_{L(w) :=} = \underbrace{3d \cdot K(w) \cdot (1 - K(w))}_{R(w) :=}. \end{aligned}$$

More derivatives:

$$L'(w) = \frac{-2d}{(1 + dw)^2} \tag{5}$$

$$L''(w) = \frac{4d^2}{(1 + dw)^3} \tag{6}$$

$$R'(w) = 3d\Big(K'(w) - 2K(w)K'(w)\Big) = 3dK'(w) \cdot (1 - 2 \cdot K(w)). \tag{7}$$

$$\begin{aligned} R''(w) &= 3d\Big(K''(w) - 2 \cdot K''(w) \cdot K(w) - 2 \cdot K'(w) \cdot K'(w)\Big) \\ &= 3d\Big(K''(w) \cdot (1 - 2K(w)) - 2 \cdot K'(w) \cdot K'(w)\Big) \end{aligned} \tag{8}$$

More facts:

- $L'(w) < 0$, $L''(w) > 0$, Equation (5) and (6).

- $L(w)$ is strictly monotonously decreasing from $L(0) = 1/d$ to $L(1) = 1/(1 + d) - d/(1 + d) > 0$ and convex.

- $R(0) = R(1) = 0$.

- Let w_0 be uniquely determined by $K(w_0) = 1/2$. (Recall that $K(w)$ is strictly increasing from 0 to 1.) We have $R'(w) > 0$ for $w < w_0$ and $R'(w) < 0$ for $w > w_0$ by Equation (7).

- $R(w)$ has one extremum: a maximum at w_0.

- For $w \geq w_0$ $R(w)$ is concave by $K(w) > 1/2$ and Equation (8).

By the first observation of (4) the first local extremum of $H(w)$ must be a minimum. In the next paragraph we show that we have at most 2 arguments w, $0 < w < 1$ with $L(w) = R(w)$. The last observation of (4) implies that $H(w)$ has at most 2 local extrema for $0 < w < 1$.

We proceed by a case distinction on $R(w_0)$ and $L(w_0)$.

- Let $R(w_0) > L(w_0)$. We have exactly one $w < w_0$ with $L(w) = R(w)$ as $L(w)$ is strictly decreasing and $R(w)$ is strictly increasing for $w < w_0$. We have exactly one $w > w_0$ with $L(w) = R(w)$ as $L(w)$ is decreasing and convex, whereas $R(w)$ is decreasing but concave and $L(1) > R(1) = 0$.

- Let $R(w_0) = L(w_0)$. We have no $w < w_0$ with $L(w) = R(w)$. We can have only one or none $w > w_0$ with $R(w) = L(w)$ as $R(w)$ is concave and $L(w)$ convex.

- Let $R(w_0) < L(w_0)$. We have no $w < w_0$ with $L(w) = R(w)$ and either none or one or two (but not more) $w > w_0$ with $L(w) = R(w)$. The argument is as before (concave vs. convex.)

We finally need some concrete values. Our calculations are made with Maple. We show only the first 5 digits after the decimal point. We assume that the first 4 are right.

Proof of (a). We consider $d = 0.614$ and $3c = 3d/2 = r = 0.921$. We have

$$L(0.602) = 0.46025 > 0.45900 = R(0.602)$$

$$L(0.604) = 0.45894 < 0.45921 = R(0.604)$$

With the observations in (4) we see that $H'(0.602) < 0$ and $H'(0.604) > 0$ and $H(w)$ has a local minimum for a $0.602 < w < 0.604$.

We show that $H(w) > 1$ for all $0.602 < w < 0.604$. Therefore we return to

$$F(w) = \exp\left(d \cdot \Big(-(5/2) + w + 3K(w) - (3/2) \cdot (K(w))^2 \Big)\right).$$

To get lower bound for $F(w)$ for all $0.602 < w < 0.604$ we substitute the first two w's in the right hand side of $F(w)$ with 0.602 and the last with 0.604. Recall that $K(w)$ is increasing in w. We get

$$\exp\left(d \cdot \Big(-(5/2) + \ 0.602\ + 3K(0.602)\ -\ (3/2)\cdot (K(0.604))^2\Big)\right)$$
$$= \quad 0.60440\ > 0.604$$

and $F(w) > 0.604 \geq w$ for all $0.602 \leq w \leq 0.604$.

There is another w with $H'(w) = 0$. For this we have $0.8 < w < 0.85$. This need not concern us as it must be a local maximum. We have no more extrema.

Proof of (b). Consider the case $d = 0.62$ and $r = 0.93$. Then $F(0.605) = 0.6494 < 0.605$ and $W < 0.605$.

4.2 Proof of Proposition 8

Recall

Proposition 8. *For the random instance $F(V, m)$ with $m = rn, r \leq 1$ we have:*
(a) $E[X] \geq (3/2)^{n(1-\varepsilon)}$ for any constant $\varepsilon > 0$.
(b) $E[X^2]\ \leq C \cdot (E[X])^2$ for an appropriate constant C.

Proof of (a). Stirling's formula yields:
$\mathrm{E}X\ =\ \sqrt{2/(\pi n)}\,(2\cdot(3/4)^r)^n\ >\ (3/2)^{n(1-\varepsilon)}$ as $((3/4)^r \geq 3/4$ as $r \leq 1)$
Proof of (b). Given 2 assignments a, b with overlap $2l = \alpha n$, that is we have $2l$ variables which have the same truth value under both a and b, the probability that a random clause is satisfied by both a and b is $=\ (3\,/\,4)\cdot(1-\alpha\cdot(1-\alpha))$. This can be seen by elementary consideration and implies that

$$\mathrm{E}\ [X^2]\ =\ \sum_{(a,b)} \mathrm{Prob}[X_a = 1 \text{ and } X_b = 1]\ =$$
$$= \binom{n}{n/2}\cdot\sum_{l=0}^{n/2}\binom{n/2}{l}^2\cdot\left(\frac{3}{4}\cdot\left(1-\frac{l}{n/2}\cdot\left(1\ -\ \frac{l}{n/2}\right)\right)\right)^m. \quad (9)$$

With $\phi(\alpha) := (3/4 \cdot (1 - \alpha \cdot (1 - \alpha)))^r$ we get for the sum of (4) as $m = rn$

$$S_n := \sum_{l=0}^{n/2} \binom{n/2}{l}^2 \cdot \left(\phi\left(\frac{l}{n/2}\right)\right)^n. \tag{10}$$

We apply the next Lemma with $q := 2$, $t := n/2$, $z := l$ to S_n.

Lemma 19. (Laplace Lemma [1]) *Let $\phi(\alpha)$ be a positive, twice-differentiable function on $[0, 1]$ and let $q \geq 1$ be a fixed integer. Let $t = n/q$ and let*

$$S_n := \sum_{z=0}^{t} \binom{t}{z}^q \phi(z/t)^n \quad \text{and} \quad g(\alpha) := \frac{\phi(\alpha)}{\alpha^\alpha (1 - \alpha)^{1-\alpha}}$$

where $g(\alpha)$ is defined on $[0, 1]$ and $0^0 := 1$.

If there exists $\alpha_{\max} \in (0, 1)$ such that $g(\alpha_{\max}) =: g_{\max} > g(\alpha)$ for all $\alpha \neq \alpha_{\max}$ and $g''(\alpha_{\max}) < 0$, then there is a constant $C = C(q, g_{\max}, g''(\alpha_{\max}), \alpha_{\max}) > 0$ such that for all sufficiently large n we have $S_n < C \cdot n^{-(q-1)/2} \cdot (g_{\max})^n$.

We get from (10) and the Laplace Lemma that $S_n \leq C \cdot (1/\sqrt{n}) \cdot (g_{\max})^n$. From Stirling's formula and (9) we get

$$\mathrm{E}[X^2] \leq 2^n \cdot \sqrt{\frac{2}{\pi n}} \cdot C \cdot \frac{1}{\sqrt{n}} \cdot (g_{\max})^n \ = \ D \cdot \frac{1}{n} \cdot (2 \cdot g_{\max})^n.$$

Below we show that $g_{\max} = 2 \cdot (3/4)^{2r}$ (see Equation (11)) and the claim holds because $\mathrm{E}X = \sqrt{2/(\pi n)} \cdot 2^n \cdot (3/4)^{rn}$ (cf. Proof of (a).)

We check that the Laplace Lemma is applicable. For the function $\phi(\alpha)$ (definition before Equation (10)) we have for $\alpha \in [0, 1]$ that $\phi(\alpha) \geq 0$. And $\phi(\alpha)$ is twice differentiable and symmetric around $\alpha = 1/2$. For $\alpha = 1/2$ we have its minimum on $[0, 1]$ which is $\phi(1/2) = (3/4)^{2r}$. (Elementary calculus for the proof.)

We come to

$$g(\alpha) = \frac{\phi(\alpha}{\alpha^\alpha (1 - \alpha)^{1-\alpha}} \ = \ \frac{(3/4 \cdot (1 - \alpha \cdot (1 - \alpha)))^r}{\alpha^\alpha (1 - \alpha)^{1-\alpha}}.$$

It turns out that $g(\alpha)$ is maximized at

$$g_{\max} \ = \ g(1/2) = 2 \cdot (3/4)^{2r}. \tag{11}$$

First, $g'(\alpha) =$

$$\underbrace{\left(\frac{3}{4}\right)^r \underbrace{(1-\alpha+\alpha^2)^{r-1}}_{=((\alpha-1)^2+\alpha)^{r-1}>0} \underbrace{\frac{1}{\alpha^\alpha(1-\alpha)^{1-\alpha}}}_{>0}}_{>0} \cdot \underbrace{\left[r(2\alpha-1)-(1-\alpha+\alpha^2)\ln\left(\frac{\alpha}{1-\alpha}\right)\right]}_{=:h(\alpha)}$$

and $g'(1/2) = 0$, as $h(1/2) = 0$. Moreover,

$$g''(1/2) = 1/3 \cdot 9^r \left(2^{-4\,r+4}r - 24 \cdot 16^{-r}\right)$$

which is easily seen to be < 0 even for $r < 3/2$.

We consider $\alpha \in (0, 1/2), r \in (0, 3/2)$. We have

$$h'(\alpha) = 2r \underbrace{- \underbrace{(2\alpha-1)}_{<0} \cdot \underbrace{\ln\left(\frac{\alpha}{1-\alpha}\right)}_{<0}}_{<2r<3} - \underbrace{(1-\alpha+\alpha^2) \cdot \left(\frac{1}{\alpha} + \frac{1}{1-\alpha}\right)}_{=:k(\alpha)}.$$

We rewrite $k(\alpha) = 1/\alpha + 1/(1-\alpha) - 1$ and

$$\begin{aligned}
k'(\alpha) &= -\frac{1}{\alpha^2} + \frac{1}{(1-\alpha)^2} = \frac{2\alpha-1}{\alpha^2 \cdot (1-\alpha)^2} < 0 \\
&\Rightarrow k(\alpha) \text{ strictly monotonously decreasing in } (0, 1/2). \\
k(1/2) = 3 &\Rightarrow k(\alpha) > 3, \forall\, \alpha \in (0, 1/2) \\
&\Rightarrow h'(\alpha) < 0, \forall\, \alpha \in (0, 1/2), 0 < r < 3/2 \\
&\Rightarrow h(\alpha) \text{ is strictly monotonously decreasing in } (0, 1/2). \\
h(1/2) = 0 &\Rightarrow h(\alpha) > 0, \forall\, \alpha \in (0, 1/2) \\
&\Rightarrow g'(\alpha) > 0, \forall\, \alpha \in (0, 1/2) \\
&\Rightarrow g(\alpha) \text{ strictly monotonously increasing in } (0, 1/2).
\end{aligned}$$

□

Conclusion

Concerning our constant r. The contribution is that the bound of $r < 0.82$ from [13] can be improved to $r = 0.921$ by more advanced techniques. Our proof cannot easily be extended beyond $r = 0.93$. In this case we have $w_{3,k} < w_{2,k} < 0.605$ by Lemma 17 (b) . Our proof for $r = 0.921$ however relies clearly on the fact that $w_{3,k} > 0.9$.

Acknowledgement. Special thanks to Anja Lau for help with the calculus.

References

1. Achlioptas, D., Moore, Ch.: Random k-SAT: Two Moments Suffice to Cross a Sharp Threshold. SIAM Journal on Computing 36(3), pp. 740-762 (2006)
2. Beame, P., Karp, R., Pitassi, T., Saks, M.: The Efficiency of Resolution and Davis-Putnam Procedures. SIAM Journal on Computing 31(4), pp. 1048-1075 (2002)
3. Broder, A., Frieze, A., Upfal, A.: On the Satisfiability and Maximum Satisfiability of Random 3-CNF Formulas. In: Proceedings 4th ACM-SIAM Symposium on Discrete Algorithms , pp. 322-330 (1993)
4. Bollobas , B.: Random Graphs. Academic Press (1985)
5. Bodirsky, M., Kára, J.: The Complexity of Temporal Constraint Satisfaction Problems. In: Proceedings 40th ACM Symposium on Theory of Computing, pp. 29-38 (2008)
6. Ailon, N., Alon, N.: Hardness of Fully Dense Problems. Information and Computation 205(8), pp. 1117-1129 (2007)
7. Christof, Th., Jünger, M., Kececioglu, J., Mutzel P., Reinelt, G.: A Branch-and-Cut Approach to Physical Mapping with End-Probes. In: Proceedings of the 1st Annual International Conference on Computational Molecular Biology (RECOMB), pp. 84-92 (1997)
8. Chor, B., Sudan, M.: A Geometric Approach to Betweenness. SIAM Journal on Disctete Mathematics 11(4), pp. 511-523 (1998)
9. Dubois, O., Boufkhad, Y., Mandler, J.: Typical Random 3-SAT Formulae and the Satisfiability Threshold. In: Proceedings 11th ACM-SIAM Symposium on Discrete Algorithms, pp. 126-127 (2000)
10. Erdös, P., Renyi, A.: On the Evolution of Random Graphs. Publications of the Mathematical Institute of the Hungarian Academy of Science 5, pp. 17 - 61 (1960)
11. Friedgut, E.: Hunting for Sharp Thresholds. Random Structures and Algorithms 26(1-2), pp. 37-51 (2005)
12. Guttmann, W., Maucher, M.: Variations on an Ordering Theme with Constraints. In: Proceedings Fourth IFIP International Conference on Theoretical Computer Science (TCS) pp. 77-90. IFIP International Federation for Information Processing 209, Springer (2006)
13. Goerdt, A.: On Random Ordering Constraints. In: Proceedings Computer Science in Russia (CSR) pp. 105 - 116. LNCS 5675, Springer (2009)
14. Isli, A., Cohen, A., G.: An Algebra for Cyclic Ordering of 2D Orientation. In: Proceedings 15th American Conference on Artificial Intelligence (AAAI), pp. 643-649. AAAI/MIT Press (1998)
15. Hatami, H., Molloy, M.: Sharp Thresholds for Constraint Satisfaction Problems and Homomorphisms. Random Structures and Algorithms 33(3), pp. 310 - 332 (2008)
16. Karp, R. M.: The transitive closure of a random digraph. Random Structures and Algorithms 1(1), pp. 73 - 93
17. Molloy, M.: Models and Thresholds for Random Constraint Satisfaction Problems. In: Proceedings 34th ACM Symposium on Theory of Computing (SToC) , pp. 209-217. ACM (2002)
18. Molloy, A.: When does the Giant Component Bring Unsatisfiability? Combinatorica 28, pp. 693-734 (2008)
19. Molloy, M.: Cores in Random Hypergraphs and Boolean Formulas. Random Structures and Algorithms 27, pp. 124-135 (2005)

20. Opatrny, J.: Total ordering problem. SIAM Journal on Computing 8(1), pp. 111 - 114 (1979)
21. Palasti, I.: On the Threshold Distribution Function of Cycles in a Directed Random Graph. Studia Scientiarum Mathematicarum Hungarica 6, pp. 67-73 (1971)